

[0265] In addition to these direct data dependencies, the TSAF data maintenance practices must manage other indirect data dependencies. For example, road characteristic data is periodically re-surveyed by generating a pre-inventory report that documents an existing route and the road characteristic data along that route. A survey crew then validates and updates the data in the pre-inventory report, and imports the newly surveyed data back into the database. The system and method must protect against changes in the definition of the route between the time the pre-inventory report is generated and when the surveyed data is imported to prevent associating road characteristic data with the wrong section of road.

10 **[0266]** The data maintenance process uses three methods to manage this complexity and ensure consistency for all data maintenance activities:

[0267] 1. session control and locking mechanisms are used that help manage all data maintenance activities.

[0268] 2. Data Dictionary is used to obtain information about the dependencies between data elements and calls to data maintenance objects (see the next item) to process any data changes, including those caused indirectly by dependencies between data elements.

15 **[0269]** 3. a collection of data maintenance objects is implemented, each of which provides data maintenance services for a particular type of data.

20

The Data Dictionary Model

[0270] The system and method are designed to provide a powerful and extensible system for storing data. In order to provide this extensibility, meta-data is maintained about most Entities and Attributes that define these Entities and Attributes to the exemplary embodiment software components. Thus, some new Entities and Attributes can be supported simply by adding new meta-data describing the new Entities and Attributes; software modifications may not be required. For example, one could add a new Division Section Attribute defining the surface type of the shoulder by providing the meta-data that describes this new Attribute. This meta-data is stored in the Data Dictionary, which is described herein.

[0271] It should be noted that the Data Dictionary is merely a repository for the properties that are defined elsewhere in this document. For example, Entity classes are characterized by properties and this definition is implemented in the Data Dictionary via an Entity Class table that lists the Entity classes and the properties of each Entity class. Thus, the definition of the Data Dictionary consists of two parts: (1) a table structure used to store and relate Data Dictionary information and (2) a list of the columns in each table and the associated properties (defined elsewhere in this document) which the column describes.

[0272] Because the Data Dictionary is a complicated object, the description of it is divided up into a series of sub-dictionaries, each documented in a separate sub-section below.

The Entities-Attributes Data Dictionary

[0273] The Entities-Attributes portion of the Data Dictionary describes the portion of the Data Dictionary that defines the meta-data necessary for the Entities and Attributes. The table structure is depicted in Figure 17, and the column definitions

5 follow:

[0274] Entity Class ([Entity_Classes]) (1700). Each row of the Entity Class table defines an Entity class by listing values for the following columns.

[0275] Description ([Descr]). This is an optional description of the Entity class, which is used in user interface screens to help a user better understand the objects that an Entity class represents.

10

[0276] Entity Class ID ([Entity_Class_ID]). Each Entity class is identified by a unique numeric ID, the Entity Class ID. This ID, along with the Entity ID, forms part of the Global Entity Identifier. This value is static once an Entity class is created and cannot be modified by a user.

15 **[0277] Entity Class Type ([Entity_Class_Typ_Cd]).** Each Entity class belongs to an Entity class type that defines the general characteristics of that Entity class. The Entity class type column contains a numeric code indicating the Entity class type for this Entity class.

[0278] Entity Class Sub-type ([Entity_Class_Sub_Typ_Cd]). The Entity class type is not always sufficient to characterize the major characteristics of an Entity class, in which case the Entity class sub-type completes the characterization.

20